



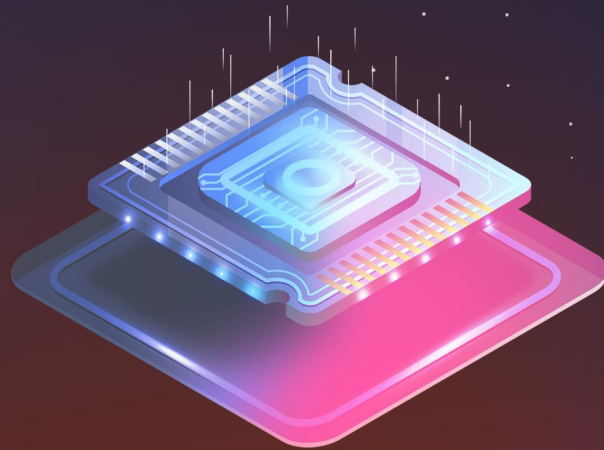
NFT Rust API

GitHub Repo:
<https://github.com/luishsr/rust-nft-api>

Project Goal

Develop a simple **RESTful API in Rust** to interact with Ethereum-based Non-Fungible Tokens (NFTs).

The API must provide functionalities for **creating, retrieving, and listing NFTs** along with their **off-chain metadata**.



High Level **approach**

Rust API

Rust-based API utilizing the Axum framework for handling HTTP requests and responses

IPFS

Off-chain metadata will be stored in a decentralized storage solution

Smart Contract

A simple ERC-721/ERC-1155 smart contract will be developed in Solidity for minting and managing NFTs

Web 3 Integration

Use Rust's web3 crate to interact with the Ethereum blockchain

Rust API overview

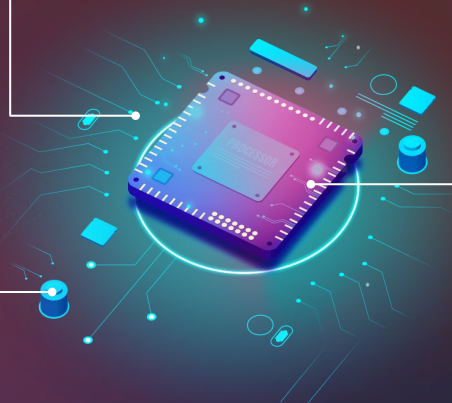
Endpoints

Mint NFT (**/mint**)

Accepts
MintNftRequest **JSON**

NFT Metadata (**/nft/{token_id}**)

Fetches NFT metadata,
including **off-chain**
data URLs, by token ID.



List NFTs

(**/tokens/{owner_address}**)

Lists all NFTs minted
through the API,
filterable by owner
address.

IPFS off-chain Data

IPFS HTTP API endpoint
(<http://127.0.0.1:5001/api/v0/add>)
for **file uploads**, enabling
interaction with a locally running
IPFS node.

Async File Handling

Opens and reads the specified file
from the **nft-images subdirectory**
within the current working directory,
converting it to bytes for **upload**.



Request Client

The request HTTP client for
asynchronous web requests,
specifically for posting the multipart
form data to the IPFS endpoint.

Response Handling

On successful upload, extracts the
IPFS hash from the response,
forming a URL
(<https://ipfs.io/ipfs/{hash}>) that
points to the uploaded content on
the IPFS network.

Smart Contract for NFT

ERC721 Extension

Inherits from OpenZeppelin ERC721URIStorage contract extension that allows for token URI storage, enabling NFT metadata association.

Defines a TokenDetails struct to store comprehensive details about each NFT, including its ID, name, owner, and metadata URI.



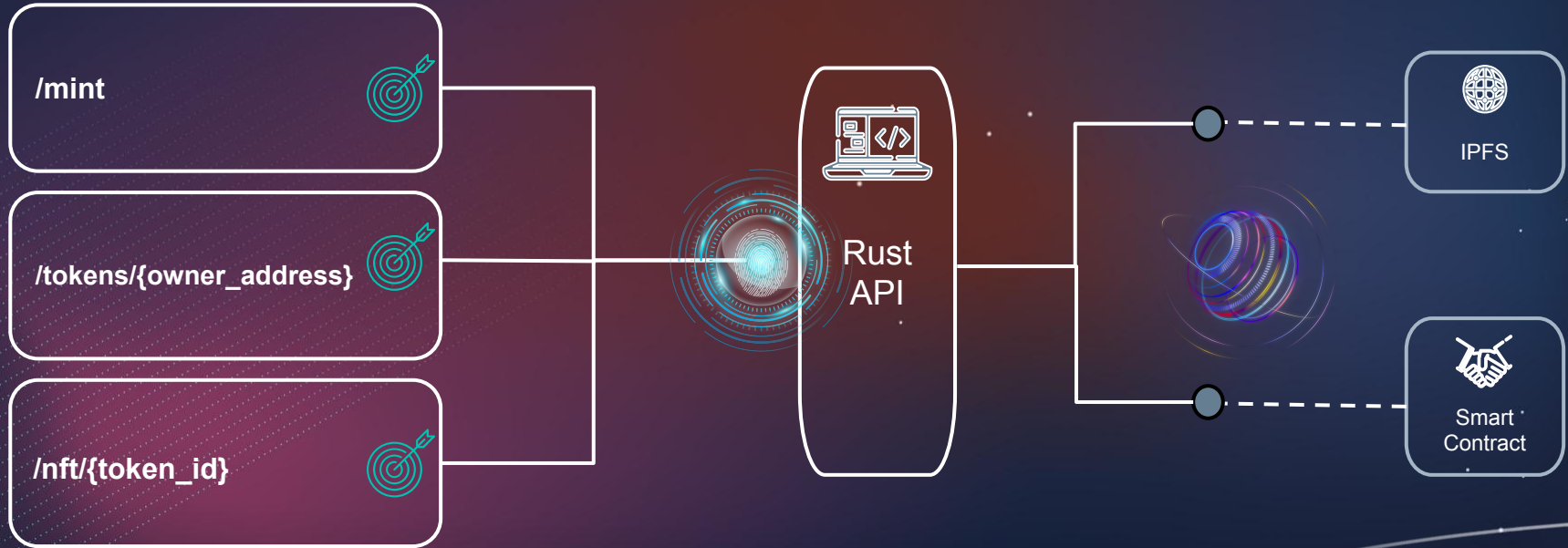
Mint NFT function mints a new NFT to a specified recipient with a given name and metadata URI pointing to IPFS hash

Token Queries

getAllTokensByOwner: returns an array of token IDs owned by a specified address/or all minted token IDs.

getTokenDetails: retrieves detailed information about a specific token by its ID, including its name, owner, and metadata URI

Workflow



Quick **demo**



Thanks!

Do you have any questions?
soaresluispt@gmail.com
+351 910 413 135

<https://medium.com/@luishrsoares>



CREDITS: This presentation template was created
by **Slidesgo**, including icons by **Flaticon** and
infographics & images by **Freepik**

Please keep this slide for attribution